



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/343,805	06/30/1999	RONNIE I. CHAIKEN	777.285US1	8333

27488 7590 01/30/2004

MERCHANT & GOULD  
P.O. BOX 2903  
MINNEAPOLIS, MN 55402-0903

EXAMINER

STEELMAN, MARY J

ART UNIT	PAPER NUMBER
----------	--------------

2122

23

DATE MAILED: 01/30/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

**Office Action Summary**

Application N .

09/343,805

Applicant(s)

CHAIKEN ET AL.

Examiner

Mary J. Steelman

Art Unit

2122

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) ☒ Responsive to communication(s) filed on 07/16/2003, 10/09/2003.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) ☒ Claim(s) 1-15, 17-25 and 27-43 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-15, 17, 18 and 31-43 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) ☒ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 30 June 1999 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. §§ 119 and 120**

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- \* See the attached detailed Office action for a list of the certified copies not received.
- 13) ☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application) since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.
- a) ☐ The translation of the foreign language provisional application has been received.
- 14) ☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121 since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.

**Attachment(s)**

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO-1449) Paper No(s) \_\_\_\_\_.
- 4) ☐ Interview Summary (PTO-413) Paper No(s). \_\_\_\_\_.
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: \_\_\_\_\_.

### **DETAILED ACTION**

1. This action is responsive to Amendment D, filed 16 July 2003 and Amendment E, mailed 09 October 2003. Claims 19 and 25 were previously amended in the Response filed 16 July 2003. Claims 1, 2, 9, 15, and 35 are currently amended. Claims 1-15, 17-25, and 27-43 are pending.

#### ***Claim Objections***

2. Claim 15, as currently amended in Amendment E, has deleted the word 'heterogeneous' from the last line. Previously the last line recited, "building an intermediate representation of heterogeneous program containing the component." Currently the last line recites, "building an intermediate representation of the program containing the components." Claim 15 should either clearly show the amended deletion or correct the inadvertent missed word.

#### ***Claim Rejections - 35 USC § 101***

3. In view of the amendments to independent claims 19 and 25, the 35 USC 101 rejections are hereby withdrawn.

#### ***Claim Rejections - 35 USC § 103***

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

5. Claims 1-3, 5-13, 15, 17, 18, and 31 – 41 are rejected under 35 U.S.C. 103(a) as being anticipated by Morgenstern, U.S. Patent 5,970,490, in view of Srivastava, U.S. Patent 5,966,539.

Morgenstern disclosed processing heterogeneous data. Col. 3, lines 7-33, "...access across heterogeneous data bases allowing integration of a wide variety of information resources including...software modules...A declarative specification language is utilized to represent the source and target data representations....In addition, a rule-like specification applies functional transformations in a manner similar to that of production rule systems...create information mediators and information bridges each of which can access heterogeneous data resources and transform that information...data is mapped to an intermediate internal format....provides reusability of each information mediator to support multiple applications..." Additionally, see Morgenstern fig. 1, where it inherently depicts a processor and memory used in the invention.

Although Morgenstern's invention does address a heterogeneous program, having components in different forms, that are transformed, Morgenstern does not provide details on basic blocks, code blocks, data blocks and determining procedures. However, Srivastava did disclose object code modules translated into an intermediate language that is compatible with a plurality of computer system hardware architectures. (Abstract, lines 5-8.) Modified code is

Art Unit: 2122

converted into executable code compatible with a target one of said plurality of computer system hardware architectures. Srivastava provided more information on using blocks and determining procedures and symbol tables.

Per claims 1, 15, 31, and 35:

Srivastava539 discloses a computer system that translates and transforms a plurality of source code modules into an intermediate representation, using a symbol table to resolve addressing issues, partitioning blocks and procedures, analyzing the flow within and between program procedures, optimizing and producing platform specific executable code (See abstract on page 1.) and comprising:

*“...obtaining a binary for each component...”* (See fig. 3.) Binaries are input as executable code.

*“...determining a plurality of basic blocks...”* (See fig. 3.) Basic blocks are displayed at 101 – 103. Srivastava539 discloses (col. 4, line 67 – col. 5, line 1), “...procedures are organized into basic execution blocks...”

*“...translating each...instruction...into an intermediate representation instruction...”* (See fig. 3.) At step 52, code is translated into “register translation language” which as stated in col. 2, lines 53 – 55, “The register transfer language is independent of any particular computer system hardware architectures.” The Examiner considers “register transfer language” to be equivalent to “intermediate representation.”

*“...determining a procedure...”* In Fig. 3 and Col. 4, lines 63 – 64 Srivastava539 discloses, “The organizer 54 partitions the linked code 52 into a collection of procedures 100.”

Art Unit: 2122

*“...creating an intermediate representation of the procedure...”* Srivastava539 also discloses (col. 5, lines 7 – 10), “...procedure flow graphs (PFG)...the PFG maps the flow of control through the basic blocks...” Examiner interprets this as mapping the flow of instructions through the blocks comprising a procedure.

*“...annotating the intermediate representation...with symbol information...”* Srivastava539 also discloses (fig. 3 and col. 5, lines 39 – 41) “The intermediate representation of the program is in the form of the register transfer language and the logical symbol table 51”

*“creating an intermediate representation of the component...”* Srivastava539 also discloses (col. 5, lines 7 – 12), “The organizer builds...a program call graph (PCG)... The PCG indicates how the procedures are called by each other.” Also, col. 11, lines 60 – 61 state, “FIG. 8 shows a program having procedures 320-329 calling each other as defined by a program call graph...”

Also see figs. 2 and 3 where code is input, analyzed and built into an intermediate representation.

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention to have modified Morgenstern’s invention to transform heterogeneous objects by including the features disclosed in Srivastava when translating objects into a neutral language because both are transforming objects, whereas Srivastava is providing more details, which are well known, regarding the steps in the transformation.

Per claim 2:

Art Unit: 2122

*“...creating an intermediate representation of the heterogeneous program from the intermediate representation of the component.”* Morgenstern, Col. 3, lines 27-28, “...the data is mapped to an intermediate internal format.”

Per claims 3, 10, and 36:

*“transforming the intermediate representation of the program based on user input.”* See Morgenstern, Fig. 1 and col. 5, lines 63-67, “The specific transformation which are compiled into the information bridge are derived from the specifications which an integration administrator provides, with this process being facilitated by some of the tools provided.”

Per claim 5:

*“... optimizing the intermediate representation of the program”* Morgenstern, col. 8, lines 12-14 and 48-52, “Schema analyzers...parse the HLDSS and create logical structure diagrams...” and “The HLDSS parser is called once to generate the code specific to the source...and called (with different parameters) a second time to process the target HLDSS.”

Per claims 6 and 7:

Morgenstern disclosed *“...Translating...into a platform-specific”*. See col. 5, lines 37-43, “This information bridge transforms data from heterogeneous data resources...into a common intermediate representation and then into a specialized target representation...”

Morgenstern does not discuss a symbol table. However Srivastava 539 disclosed *“...symbol table ...to define an address space”*. At col. 5, lines 21 – 22, “All symbolic addresses are resolved...” Also, Figure 3, item 53 shows the logical symbol table, LST. Srivastava also discloses in Fig. 3 the LST outputting information to the organizer, item 54.

Therefore, it would have been obvious to one of ordinary skill in the art, at the time of the invention to modify Morgenstern's transformation to include information regarding symbol tables because they are well known in the art and generally used in compiling code.

Per claim 8:

*"...Outputting emitted block information"* Srivastava539 also discloses in fig. 3, item 100, comprised of blocks, item 105, with an output to the code generator, item 57.

Per claim 9:

*"...obtaining the emitted block information..."* Srivastava539, See fig. 3, item 100 for stored collection of blocks, 105 in the form of intermediate representation.

Per claim 11:

*"...replacing ...platform-specific instruction with a platform-neutral..."* Morgenstern, Col. 5, lines 37-43, "...transforms data from heterogeneous data resources...into a common interface..."

*"...replicating a complex platform-specific instruction in an intermediate representation..."* Morgenstern, Col. 11, lines 30-32, "Logical structure diagrams provide a form of meta-schema in that LSD's may be used to graphically describe different data models and schemas."

Per claim 12:

*"...intermediate representation of the program arranged in a hierarchy..."*  
Morgenstern, col. 11, lines 24-30, "...HLDSS is parsed and processed by schema analyzers...to create annotated logical structure diagrams (LSDs)...each of which is a schematic structure



Art Unit: 2122

graph (like a parse tree) (a hierarchy) that represents the schema and data structures of the data resources.

Per claim 13:

*“...code block element references...instruction for multiple instances of a platform-specific instruction...”* Srivastava, fig. 3. A code block as shown in item 100 references intermediate representation instructions for multiple instances of platform specific instruction. Later in the processing, the CPU Architecture Description, item 9, works with the Code Generator, item 57, to make code platform-specific.

Per claim 17:

*“...analyzing the binary...performing code discovery...determine a plurality of basic blocks...establishing block relationships...”* See response above to claims 1 and 15, as disclosed by Srivastava539. Also see fig. 4 and col. 9, lines 37 – 39 in which Srivastava539 discloses, “The basic blocks 151-155 of the procedures 101-103 of the program 20 are repeatedly examined during successive passes...”

Per claim 18:

*“...building the intermediate representation ...comprises translating every instruction...into an intermediate representation...”* Morgenstern, col. 11, lines 33-35, “LSD internal representation is that a context independent uniform graph-based representation can represent all anticipated data resource schemata and structures.”

Per claims 32, 33, and 34:

*“Application program interface instructs the...system to further cause the processing unit to transform the plurality of intermediate representation instructions.”*

Art Unit: 2122

*“...translation and transformation system further causes processing unit to translate...into a modified platform-specific binary.”*

*“...processing unit to translate the modified platform-specific binary into a modified plurality of...instructions...”*

Morgenstern, col. 3, lines 1-6, “An information bridge is created with the interoperability assistant module through a process of program generation and the source data is processed through the information bridge to provide target data...”

Per claims 37, 38, 39, and 40:

*“...translating the plurality of intermediate representation instructions...into a modified platform-specific binary”*

*“...translating the modified platform-specific binary into a modified plurality of intermediate representation instructions for further transactions.”*

*“...translating ...instructions into a new version of the platform specific binary.”*

Morgenstern, col. 15, lines 15-23, “Each of those rules thus serves to coordinate and transform some input or intermediate data objects to other intermediate objects or output objects...The collection of rules...maps the input structure of schema to the output structure/schema. Execution of the rules in the information bridge then carries out the actual data transformations...”

Per claim 41:

*“...iterating an intermediate representation of a heterogeneous program...create a plurality of new versions...”* This is an apparatus version of the claimed methods discussed above, wherein the claim limitations also have been disclosed and/or covered under the same

cited areas as set forth above. Multiple executions through the system can be used to create a plurality of new versions of the heterogeneous program. Thus, the same rationales provided in the rejections of the above claims is also applied and incorporated herein. Additionally, see Morgenstern, col. 7, lines 16-23, “The IA (Interoperability Assistance Module) consists of several internal modules...two high level data structure specification and a high level transformation rule specification...The HLDSS data representation is reusable for all applications which need to access this data resource, for either input or output and may be edited as desired.”

6. Claims 4, 42, and 43 are rejected under 35 U.S.C. 103(a) as being unpatentable over Morgenstern, U.S. Patent 5,970,490, in view of Srivastava, U.S. Patent 5,966,539, and further in view of Srivastava, U.S. Patent 5,539,907.

Morgenstern disclosed transforming heterogeneous objects, Srivastava539 disclosed more specific information, adding transformation details regarding intermediate representation of code. Morgenstern shows user input in fig. 1, but fails to mention instrumenting. However, Srivastava907 discloses a translation program that includes instrumentation and monitoring by factoring in user input parameters. Srivastava907 discloses, (Abstract, lined 10 – 12) “Fundamental instrumentation routines identify, locate, and modify specific program components to be monitored.” Also, (col. 6, lines 62 - 64) teach, “The user supplied UIR [user instrumentation routines]...locate and modify specific portions ...”

It would have been obvious to one skilled in the art, at the time the invention was made, to modify the code transformation teachings of Morgenstern and Srivastava539, by providing the

Art Unit: 2122

teachings of Srivastava907 in order to provide a more useful customized translation system, because it allows the system to handle program transformations while the user concentrates on what performance data are to be collected through instrumentation, and how the performance data are to be analyzed. Instrumented performance data are communicated for analysis by simple fast procedure calls, reducing the monitoring overhead.

As per claims 42 and 43:

*“...manipulating the intermediate representation using data input...” and “terminating the iterating...based on data input...”* Morgenstern disclosed manipulations via rules, input controlling of the transformation and translation process, supplied through user inputs. See fig. 1. Srivastava539 disclosed additional information regarding blocks in the translation process. Neither discussed instrumentation and monitoring.

However, Srivastava907 discloses a translation program that includes instrumentation and monitoring by factoring in data input. See figs. 3 and 4. Srivastava907 (col. 6, lines 66 - 67) teaches, “The user supplied UAR 49 are procedures for collecting and analyzing performance data.”

Thus, it would have been obvious to one skilled in the art, at the time the invention was made, to modify the teachings of Srivastava539, by using data input, to the one or more iterations, and termination, of translation and transformation of platform-specific heterogeneous programs, using the Srivastava907 reference, in order to increase optimization, because data inputs can be used to alter the code sequence for the purpose of ultimately producing faster platform-specific executable code.

7. Claim 14 is rejected under 35 U.S.C. 103(a) as being unpatentable over Morgenstern, U.S. Patent 5,970,490, in view of Srivastava 5,966,539, as applied to claim 13 above, and further in view of common knowledge in computer art.

Per claim 14:

“...associating a hash value ...so that...instructions hash to the hash value.” Breslau and Srivastava539 do not explicitly address hash values. Official notice is taken that a hash table and associated hash values are well known in the computer art as a common means of indexing into a collection of data and/or code.

It would be obvious to one skilled in the art at the time the invention was made to also utilize a hash table feature to position instructions in a collection allowing quick access by an index value, for the purpose of streamlining program organization.

#### ***Response to Arguments***

8. Applicant's arguments with respect to claims 1-15-17-18, and 31-43 have been considered but are moot in view of the new ground(s) of rejection.

#### ***Allowable Subject Matter***

9. The following is a statement of reasons for the indication of allowable subject matter: As Applicant has pointed out on page 11, of Amendment E, independent claims 19 and 25 have been amended to reflect execution of instruction and accomplishment of a functional objective.

Art Unit: 2122

As such the prior rejections have been overcome. Thus dependent claims 20-24 and 27-30 also constitute allowable subject matter.

### ***Conclusion***

10. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

A search update provided the following related patent reference:

US Patent 5,175,856 to Van Dyke et al. See figure 1, which show heterogeneous components being provided to the transformation module.

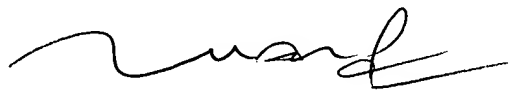
11. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Mary Steelman, whose telephone number is (703) 305-4564. The examiner can normally be reached Monday through Thursday, from 7:00 A.M. to 5:30 P.M. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Dam can be reached on (703) 305-4552.

The fax phone number is (703) 872-9306 for regular communications and for After Final communications. Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703) 305-3900.

Mary Steelman



01/23/20004



**TUAN DAM**  
**SUPERVISORY PATENT EXAMINER**